Benefits of CI/CD to achieve, build and deploy automation for cloud-based software products

Presented by Amos Nyaburi

Continuous Integration and Continuous Deployment (CI/CD) is the technical practise used in automating the process of building reliable software products. It is used to integrate, build, test and deploy code automatically from code to production

Some of the benefits of using CI/CD include:

- Smaller code changes are simpler: When it comes to updating of codes, with the help of CI/CD. It significantly reduces the time spent and resources on pushing new updates. If for example, you were to pay a developer $300 for code updates and deployments. With the use of CI/CD, you can reduce the cost to $200. Because the developer won't need to manually deploy the updates
- Better Code Quality and ensures Faster Time to Market: unit and integration test needs to be carried out before auto deployment can take place, which leads to better code quality and reduction in the shipping of buggy production code, which will save you some significant amount of money

Benefits of CI/CD (Cont)

- Increased Focus on Core Responsibilities: With CI/CD, you can confidently focus your revenue in other section of the company, because you’re certain that they won’t be any issue with deployments, and even if there is, you’ll get an immediate notification of what went wrong and where. Thereby saving you money from hiring QA testers.

The cost of CI/CD

Because you are introducing automation to the system, CI/CD pipeline implementation will be expensive. Costly operations, rollouts, and training are the possible reasons for the rise in cost. Whether you use a SaaS or manage it yourself, these expenses will still be incurred. The up-front charges must be considered first. To reduce typical errors and increase the effectiveness of your CI/CD budget, there are also hidden expenditures that you should be aware of. Therefore it is better to start CI/CD immediately when you start any project in order to reduce the cost of implementation, because it is easier to automate at the start of the project than at the end of the project. Without CI/CD you'll need to pay QA testers to test the project which will incur an extra cost and time to the timeline of the project.

The end